Efficient and Robust Optical Character Recognition Algorithm for Signature Recognition

Wafa Elmannai, Khaled Elleithy, Varun Pande
Department of Computer Science and Engineering
University of Bridgeport
Bridgeport, CT 06604, USA
{welmanna, elleithy, vpande}@bridgeport.edu

ABSTRACT

With the technology development over the past decades, it became necessary to provide secure recognition systems. The Optical Character Recognition (OCR) can be considered as one of the most useful software to offer security. It works on the principal of recognizing the patterns with the use of a computer algorithm. OCR has multiple uses in places that need security verification such as banks, elevators, police departments. Furthermore, it can be used in several categories simultaneously. There are two types of recognition. First is the static approach which is based on the information of the input. Second is the dynamic recognition which is more usable for recognition of speech. In fact, OCR will be one of the most important techniques for human computer interaction in future. However, in this paper we have used OCR as feature to implement our algorithm. We are presenting a new algorithm that is capable of recognizing each signature individually. This makes the system more efficient and robust,especially in banks which need to verify the customer's signature on a regular basis. A highly efficient C# system was developed to implement the new algorithm.

Keywords: OCR; Signature Recognition; Hand writing recognition.

1. INTRODUCTION

Since 1950s, OCR has been motivating many researchers. It has become the most useable software for automatic recognition. In fact, the first version of OCR had emerged in middle of 1940s as software package. But in middle of 1950s, OCR devices were available in the market. Nowadays, both OCR's hardware and software have become available in the market [1].

As seen, in some applications there are income prehensile characters or symbols that are used. This can be a security protocol in some systems or is just a way to track goods. Most of the products have serial of numbers or symbols. Each symbol or character has meaning [7]. A custom based OCR is the only software which has the ability to recognize such symbols. OCR can also be used in the recognition of poor hand writing.

There are several pending issues that need to be addressed to improve the OCR model. The goal of the OCR is to transform any text document to another format (e.g. PDF). Also, it is used in post offices as well as in recognition systems [2].

To have a proper recognition of a character, we need to follow three steps. These steps are segmentation, classification and extracting. The most primary and important step is the segmentation [5]. Segmentation separates the input to individual characters [4]. Also, the process of segmentation includes three sub steps. They are character, line and word segmentation. Before we explain the purpose of our implementation, we are going to explain the process of OCR.

The process of classification: in order to build a classifier to any application, there are two steps that are needed to be followed. They are the data training and data test usages. Both training and testing are mostly similar in their process. There is only one difference which is one extra step need to be done in testing procedure. First, we need to put the input or scanned image in as an accepted form. Then we extract the data to the point where we can reach the needed result. Lastly we choose the most suitable model per class. In testing step, instead of estimating the satisfactory model for the data, we need to make a comparison between the several paths and the chosen models.

Second step is the preprocessing: first thing in the preprocessing is the binarization that is the conversion of the images to bit maps. This is used to choose the threshold level based on the level of gray of the input. Second sub step is the morphological operating [3] to keep only the edges of each character. Finally, the segmentation is to separate the characters individually is based on many approaches that have been introduced such as: connected components extractor, threshold sub pixel – precise thresholding [6].

Third step is feature extraction: this process is based on the separated characters given by the segmentation step. There are several approaches to get the descriptor of the image. These approaches can be used either on the entire image or as an object to use the character's edges for the recognized step. This process is the most important feature in OCR. It allows us to improve the performance of OCR.

The work presented in this paper is not OCR. But we are using OCR as a feature in our implementation. Each implementation needs a specific classifier. So, in this paper we are going to develop a new approach in the classification process in order to recognize each individual signature as well as to recognize the hand writing specially the poor ones. This approach is based on the predefined classification process of the OCR. Our new approach can improve the performance of pattern recognition, signature verification and security enhancement. This implementation can be used for secure systems such as banks which need to verify the signature of each customer carefully. Thus, our system will enable the whole verification process to be automated.

This paper starts by explaining the framework of our approach. Then it provides the details on how combining the OCR feature in our work for signature recognition. Finally, we analyze the experimental results.

2. RELATED WORK

In [2] the author has used a similar segmentation process that has been described in our paper. Though the segmentation is done on the bases of separating lines and running digital string recognition systems. At times it can be difficult to recognize certain patterns of handwriting as people tend to write differently, thus in our case a data base with certain trained characteristics helps us improve upon the detection of one's individual handwriting .

In [8] the word recognition system using the lexicon model approach has been discussed. The author has talked about multiple segmentation hypotheses on mathematical model for the recognition of individual characters. The mathematical model describes points of weight in each individual characters thus deducing the points based upon the cusp and curve of an individual character.

In order to use OCR for Latin handwriting recognition, the authors have used a semi Markov model plus the character template set [9]. The unnoticed characters were read using a lexicon model. The algorithm was clear and understandable by using multiple samples. On Other hand it was possible to improve the idea to be a multilingual system.

The authors in [10] have combined three techniques to recognize handwriting samples. The techniques are the lexicon model, n_gram statics and the implementation of a font database. This system could easily correct ligatures and corrects the errors that were sufficient. But they used only two sample images which are not enough to evaluate the recognition system.

3. SIGNATURE RECOGNITION PROCESS

3.1. Trained data usage for classification process:

Basically this algorithm is developed to give a high performance of signature recognition.

Figure (1) shows the process of recognition and how we combine OCR feature with our algorithm to improve the process of pattern recognition and having good results.

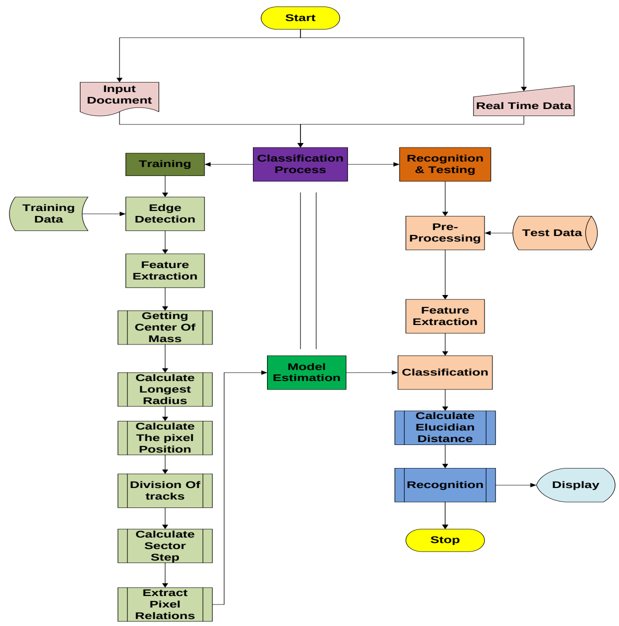

Figure 1: The signature and hand writing recognition flow chart

The input can be either from a scanner or a camera. This software can offer high security. Moreover, it recognizes each signature individually with less error. All these inputs have to be stored in the database for each client and is recalled whenever we do verification. This does not only allow for the algorithm to learn more but also provides a future implementation that enables for better understanding and achieving optimal results.

In order to build the classifier we have to follow several sub steps:

1. Training Data usage: we are trying to learn the features of the new input data (signature, new hand writing or even just a character).
2. Training:The process begins with (edge detection) Segmentation where it extracts features from each individual character
3. Computation or extraction of features from the data base which is done by the following sub steps:
    A. Compute the pixels numbers in each input character segmented.
    B. Getting the center of mass, i.e. most pixels at a point.
    C. Oval factors need to be considered :

I. Conventionality.
II. Position.

D. Kurtosis and skewness.
E. Moments of high demand.
F. A series of code conversion.
G. Series and Fourier conversion.

3.2. The Recognition Process:

As shown in Figure 1 after we finish the classification process the output of classification process is fed to the recognition process in order to test the data and check the performance of the testing process of recognition of the data. This process has been done for real time data.

3.2.1. Getting the center of frame:

We have used two parameters in our equation. We have used Figure(2) as input to calculate Vb. That is to get the center of our input. We have used Vb as the sum of all the locations divided with the pixels number.

Vb=$\sum$v / $\sum\sum f$ (v, h)

	0	1	2	3
1				
2			*	*
3		*		
4	*		*	

Figure 2: Getting the central of textual (Vb)

To have the result mathematically, we can calculate the central for Vb= (0+1+2+2+3)/5.

And the same process is done to calculate Hb. We have used the Figure(3) as our input to get the central of Hb.

Hb= $\sum$h/ $\sum\sum f$ (v, h

	0	1	2	3
1				
2			*	*
3		*	*	
4			*	

Figure 3: Getting the central of textual (Hb)

The Hb for out input is (3+2+2+2+1)/5.

If we apply this equation of our algorithm we can easily get the central of the textual as showing in Figure (4).

Following this process, we can get the mass central for each character in the signature.

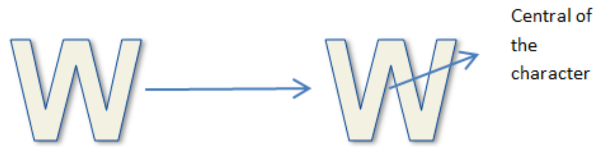

Figure 4: Central of Character

3.2.2. Getting the highest range:

To find the highest radius, we have used Pythagoras operation. This operation allows us to find the distance between the central of character and each pixel in the character. The largest distance will be considered as the highest radius as shown in Figure (5).

To measure any distance between two pixels with use of Pythagoras, we its equation which is as following: $sqrt\ ((VPc - Vpi)^2 + (HPc - Hpi)^2)$

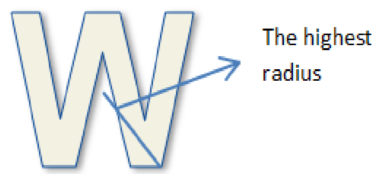

Figure 5: Highest Radius

3.2.3. Find the step of track:

To get the track numbers, we have to divide the highest radius over the Predicted tracks number. However, we have used five as the number of tracks.

3.2.4. Find the right sector (depend on sectors numbers):

For this step we have to the number of sectors in our input based on the tracks number. Based on finding the step of the sector, we can know the specific sector for each pixel.

3.2.5. Dividing the sectors numbers over the tracks number:

Both data (trained or tested) has to use the same numbers of tracks and sectors to specify the pixel position. Figure (6) shows how to find the right position of specific pixel.

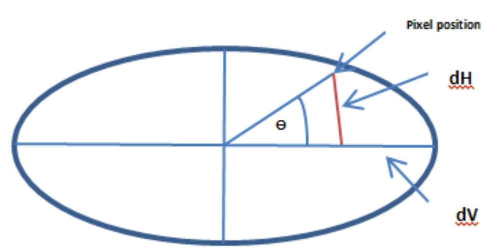

Figure 6: Target for the pixel position

$\theta i = tan - 1(H - Hc/v - Vc)$
The pixel position can be found based on
$$Target\ sector = \frac{\theta}{stepof\ sector}$$

3.2.6. Getting the relationship between each pixel and their neighbors:

This step is the last step of feature extraction. We have used chain code with some modification:

```
Start
1. Adjacent the target pixel for each pixel in the
 character.
2. Changing clock-wise from north path
3. Whenever the current pixel presents surrounding
 the target pixel
Then
        3.1. Save its position in the database
        3.2. Go to next pixel
4. End the loop
5. Next
   END
```

Using the chain code, we can find the right position of the pixel and its relationship with other pixels.

3.3. Euclidian Distance Calculation:

This step can be considered as classification process for the real time data. In this code we could solve the chain code drawbacks of calculating the distance. With the distinction of multiple tracks number and sectors we could easily calculate the actual distance and determine the relationship between the vectors.

4. EXPERMENTS AND RESULT ANAYSIS

With the improved algorithm of the freeman's chain code and the initiation of the Fourier transform series, we solved the scaling problem as already shown in the paper by considering individual vector paths for the pixels. We have also achieved accuracy percentages as shown in the pie chart shown in Figure (7). These results are based on the handwritten test data we used were of two different types.

In our testing, we asked volunteers to draw the characters using the paintbrush as they would write on a piece of paper. As a second step, we made the volunteers write the same letters on a piece of paper with a pen. Once this was done, sample signatures of the same volunteers were fed into the classifier and were trained on an individual's data. We had approximately 20 samples of 17 different hand written letters and 20 signature samples per volunteer for the best accurate results.

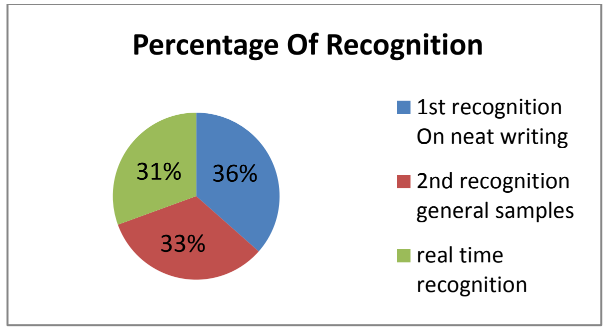

Figure 7: Achieved Accuracy Percentages for Handwritten data

For our experiment, we chose one sample of the stored handwritten samples and download it in our system as shown in Figure 8(a).

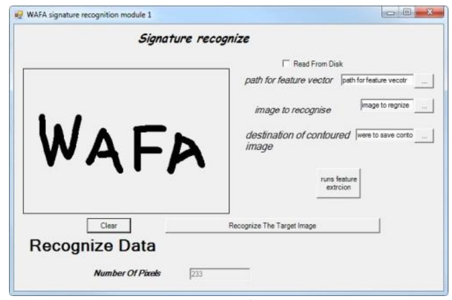

Figure 8(a): Sample of handwritten data

Thresholding setup was used for the segmentation process in our experiment. The result of feature extraction which was done on the sample is shown in Figure 8(b).

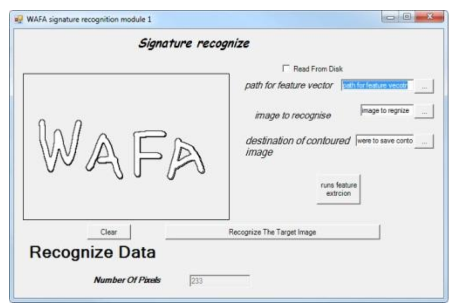

Figure 8(b): Recognition approach of the sample

As seen in the Figure 8(b), using of the technique robust gray value in the feature extraction is showing good results of extraction and a purer outline. As well for classification process, we have used binarization algorithm on the sample. Based on our previous experiment for handwritten recognition, we have used one signature of the signature samples which is shown in Figure 9(a). We have used same approach to recognize the signature.

The result of extraction of features is shown in Figure 9(a). The extraction was done on the whole image then gives the number of pixels as shown in the above figure.

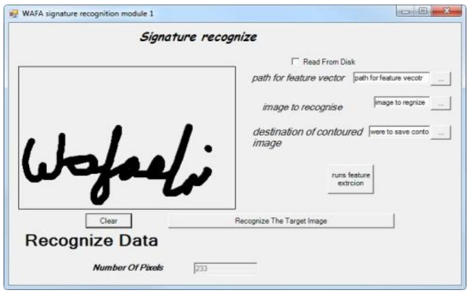

Figure 9(a): Sample of Stored signature

Figure 9(b) shows the recognition technique on both samples. Furthermore, our approach proves the way of recognition was correct for all the characters.

Figure 9(b): The recognition technique presented on both samples.

There are two main observations that we noticed in our experiments which can affect the recognition technique.

First, some volunteers used different types of fonts that we have to use based on our classification. Others, proportioned symbols with the bending characters can be hard to recognize especially for poor handwritten.

5. CONCLUSION

Using OCR in our system to recognize both the signature and handwrittencharacters was very helpful. We have developed C #based system character recognition. The developed system supports both segmentation and classification in an efficient and robust form.

In this paper we have used a feature extraction process of OCR to achieve a highly efficient and robust recognition technique. Furthermore, we have used both robust gray value and dynamic thresholding techniques in the preprocessing (segmentation plus feature extraction). Such an approach makes the developed system more practical not only the classifier gets trained for the individuals' handwriting but also gets trained for the lower case, upper case as well as the signature of an individual's data which helps the classifier achievemore accurate results. Furthermore, the algorithms presented in this paper offer better security vital systems such as banks which need to verify the customers' signatures. Finally, the developed system proves to achieve accurate recognition results even for poor handwritten.

6. REFRENCES

[1] Eikvil Line, "Optical Character Recognition, Norsk Regnesentral," P.B. 114 Blindern, N-0314, 1993.

[2] Sargur N. Srihari and Sangjik Lee, "Automatic Handwriting recognition and Writer Matching on Anthrax-related Handwritten Mail", Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR'02), 2002.

[3] Luo Huizhu, Kasturi Rangachar, "Improved Directional Morphological Operations for Separation of Characters from Maps/graphics, in: Graphics Recognition Algorithms and Systems [M]," Heidelberg: Springer Berlin Press, 2006.

[4] Sagar.B.M.,Shobha.G, Kuma.R.P, "Complete Kannada Optical character Recognition with Syntactical Analysis of the script," Computing, Communication and Networking, 2008. ICCCn , 2008.

[5] Sari Toufik, Souici Labiba, Sellami Mokhtar, "Off-line Handwritten Arabic Character Segmentation Algorithm: ACSA," IEEE Explore, 2009.

[6] Paidimarri Kashyap, Chandran Sharat, "Ad-Hoc Multi-planar Projector Displays, in: Computer Vision, Graphics and Image Processing[M]," Heidelberg: Springer Berlin Press, 2006.

[7] Xie Jianhong, "Optical Character Recognition Based on Least Square Support Vector Machine," 2009 Third International Symposium on Intelligent Information Technology Application, 2009.

[8] E. Anquetil, G. Lorette, "Perceptual Model of Handwriting Drawing Application to the Handwriting Segmentation", Problem Pattern Analysis and Machine Intelligence, IEEE Transactions Jan 2000

[9] Shou de Lin and Kevin Knight. Discovering the linear writing order of a two-dimensional ancient hieroglyphic script. Arterial Intelligence, 170: to appear, 2006.

[10] Chi Fang and Jonathan J. Hull. A modified character-level deciphering algorithm for OCR in degraded documents. In SPIE Conference on Document Recognition II, 1995.